

PUB-NO: JP360017539A
DOCUMENT-IDENTIFIER: JP 60017539 A
TITLE: EMULATION SYSTEM

PUBN-DATE: January 29, 1985

INVENTOR-INFORMATION:

NAME	COUNTRY
ADACHI, SHIGEMI	
NAKAOSA, YOSHITAKE	
FUJIOKA, YOSHINORI	

ASSIGNEE-INFORMATION:

NAME	COUNTRY
HITACHI LTD	

APPL-NO: JP58126322
APPL-DATE: July 11, 1983

US-CL-CURRENT: 422/28
INT-CL (IPC): G06F 9/44

ABSTRACT:

PURPOSE: To emulate the input and output instructions efficiently by a small quantity of the overhead, by installing the means, etc., by which input and output control macro instructions issued from the user program are judged.

CONSTITUTION: The means, etc., which will judge the input and output control macro instructions issued from the user program by the supervisor call instructions, are installed. For example, an interface program ECP21 is installed between the operating system 10 of the native machine and the program 22 of the target machine. And, to the program 22 emulated by ECP21, EXL instructions, which will directic the mode switching, are issued, and emulation is started. Next, the input and output control macro instructions issued in the user program 24 of the target machine are converted to the input and output control macro instruction to the operating system 10 of the native machine.

COPYRIGHT: (C)1985,JP0&Japio

⑬ 日本国特許庁 (JP)

⑪ 特許出願公開

⑫ 公開特許公報 (A)

昭60—17539

⑤ Int. Cl.⁴
G 06 F 9/44

識別記号

庁内整理番号
A 7361—5B

④ 公開 昭和60年(1985)1月29日

発明の数 1
審査請求 未請求

(全 6 頁)

⑤ エミュレーション方式

式会社日立製作所システム開発
研究所内

① 特 願 昭58—126322

② 発 明 者 藤岡良記

② 出 願 昭58(1983)7月11日

尾張旭市晴丘町池上1番地株式
会社日立製作所旭工場内

⑦ 発 明 者 足立茂美

① 出 願 人 株式会社日立製作所

尾張旭市晴丘町池上1番地株式
会社日立製作所旭工場内

東京都千代田区神田駿河台4丁
目6番地

⑦ 発 明 者 中蔵恵丈

④ 代 理 人 弁理士 磯村雅俊

川崎市麻生区王禅寺1099番地株

明 細 書

1. 発明の名称 エミュレーション方式

2. 特許請求の範囲

(1) 命令語体系の異なるデータ処理装置のプログラムをエミュレーションするマイクロプログラム制御のデータ処理装置において、被エミュレーション・プログラム中のスーパーバイザ・コール命令または直接オペレーティング・システムに刺込む命令を検出し、該命令のオペランド値により入出力制御マクロ命令か否かを判定して、入出力制御マクロ命令であるときには、エミュレーションをコントロールするプログラムに刺込みを発生し、上記以外の命令であるときには、被エミュレーション・プログラムのオペレーティング・システムに刺込みを発生することを特徴とするエミュレーション方式。

3. 発明の詳細な説明

(発明の利用分野)

本発明は、エミュレーション方式に関し、特に

頻繁に入出力命令が発行されるオンライン・プログラムのエミュレーションが可能なエミュレーション方式に関するものである。

(発明の背景)

通常のデータ処理装置を用いて問題を解くためには、その問題の処理手順をプログラムの形に記述する作業、つまりプログラミングが必要であり、このプログラミングを容易にするため FORTRAN, COBOL 等のマクロ命令が広く使用されている。しかし、マクロ命令で書かれたソース・プログラムは、そのままでは実行することができず、コンパイラ方式、インタプリタ方式あるいはコンパイラ・インタプリタ方式等により翻訳して機械語のプログラムに変換した後、実行する。

通常のデータ処理装置は、その装置に固有のプログラム、すなわち命令語群を持つており、各命令語の特定フィールドに記述された操作コードによりその命令語の機能が定められている。

ところで、新しいデータ処理装置が開発され、旧機種と入れ換えられる場合に、旧機種で開発さ

れて使用されてきたプログラムをどのようにして受け継ぐかという問題がある。その場合、ソースプログラムがマクロ命令で書かれていれば、システム切換え時に再コンパイルすることにより、新しい機種（ホータイプ・マシン）の目的プログラムとして使用することができる。しかし、現実には、それぞれのデータ処理装置の言語仕様に差異があつて、完全な互換性がないことが多く、そのために修正が必要となり、またコンパイルの後のテスト・ラン等のため、人手による作業が多く必要となる。

したがつて、新旧機種の間にそのアーキテクチャの上で互換性がないにもかかわらず、旧機種（ターゲット・マシン）のプログラムをそのままの形で新機種のデータ処理装置（ホータイプ・マシン）により実行する場合、あるいはアーキテクチャの異なる仮想計算機のプログラムを実在のデータ処理装置で実行する場合には、エミュレーションが必要となる。エミュレーションは、マイクロプログラミングを利用して新しい処理装置（ホータイプ・マシン）の機能を拡張あるいは変更し、シミュレーションを高速に行うものである。

エミュレーションを行う場合、特に問題となることは、入出力命令のエミュレーションである。

例えば、チャネル方式でない入出力命令体系を持つデータ処理装置のプログラムを、チャネル形式の入出力命令体系を持つデータ処理装置でエミュレーションするような場合には、ハードウェアの違いが大きすぎるため、マイクロプログラムのみでエミュレーションすることは困難となる。このような場合、エミュレーションを行うデータ処理装置のオペレーティング・システムの1つのパーティション（区画）に被エミュレーション・プログラムを置き、両者の間にインタフェース用プログラムを設ける方法が用いられている。このとき、マイクロプログラムは、被エミュレーション・プログラム中に入出力命令を検出すると、インタフェース用プログラムに通知し、ソフトウェアにより入出力命令をインタプリット（解釈実行）して、ホータイプ・マシンの新オペレーティング・システムを発行するのみで、入出力動作が可能である。

プログラムの入出力マクロ命令または入出力命令に置換える処理を行う。この場合、被エミュレーションプログラム中の複数の入出力命令が1つのホータイプ・マシン用オペレーティング・システムの入出力マクロ命令または入出力命令に置換えられるのが普通である。しかし、この方式では、エミュレーションを制御するインタフェース用プログラムの上に、さらにホータイプ・マシン用オペレーティング・システムが存在する形となるため、オーバーヘッドが大きくなることは避けられない。

通常、エミュレーション機能の中で、マイクロプログラムの分担する部分を多くすれば実行速度は上るが、融通性は少なくなり、逆に、ソフトウェアの分担する部分を多くすれば実行速度は遅くなるが、融通性は増大する。

一般に、チャネル形式でない入出力命令体系のデータ処理装置では、1つの入出力動作を行うために複数の入出力命令が発行される。これに対して、チャネル形式の入出力命令体系を持つデータ処理装置では、入出力起動命令（SIO命令）1

つを発行するのみで、入出力動作が可能である。

例えば、磁気ディスク装置に対してシークおよびリード動作を行わせる場合、チャネル形式の入出力命令体系では、チャネル・コマンド・ワード（CCW）にてシーク動作とリード動作をコマンド・チェーンにより指定しておけば、チャネルに対する入出力起動命令（SIO命令）を発行するのみですむのに対して、チャネル形式でない入出力命令体系では、ディスク制御装置内の入出力制御レジスタ群に対し、逐一シーク、リード動作の各々に必要なパラメータ類を精込んだ後、入出力を起動する方法がとられる。

第1図は、チャネル形式でない入出力命令体系を持つ処理装置の磁気ディスクに対する入出力命令群の説明図、第2図は第1図の1つの命令フォーマットを示す図である。

ターゲット・マシンがチャネル形式でない入出力命令体系を用いている場合、磁気ディスクに対してシーク動作を行わせるためには、第1図(a)に示すように、①の命令で磁気ディスクのユニット・

アドレスを指定し、②の命令でシークするトラック・アドレスおよびセクタ・アドレスを入出力制御レジスタに書き込み、③の命令でシーク動作のコマンドを入出力制御レジスタに書き込むと同時に、シーク動作の起動を行う。また、磁気ディスクに対してリード動作を行わせるためには、第1図(b)に示すように、①、③の命令で読み込み先の主記憶装置アドレスの上位、下位を入出力制御レジスタに書き込み、③の命令でリードするデータのワード数を指定し、④の命令でリード・コマンドを入出力制御レジスタに書き込むと同時に、リード動作の起動を行う。次に、これらの命令フォーマットについて説明する。第2図は、第1図におけるライト命令WIO (Write I/O) のフォーマットを示している。

WIO命令は、磁気ディスク制御装置内の入出力制御レジスタへのデータ書き込みおよび起動を行うものである。命令要素のWIO n (B, F), Rのうち、WIOは入出力装置への書き込みを指示するオペレーション・コード(OP)、n部は入

出力制御レジスタ番号、B部は入出力装置アドレスがセットされる汎用レジスタ番号、F部は入出力動作の起動を指定するビットで"1"のとき起動の指示を表している。R部は入出力制御レジスタに書き込まれるデータを保持する汎用レジスタ番号である。実際の命令フォーマットは、第2図のようになり、OP、B、F、n、Rの順序で配列されている。

なお、入出力制御レジスタ番号を示すn部が、(000)のときにはワード・アドレス・レジスタ(WAR)、(001)のときにはコントロール・レジスタ0(CTRO)、(010)のときにはコントロール・レジスタ1(CTRI)、(011)のときにはユニット・アドレス・レジスタ(UNR)、(100)のときにはアドレス・コントロール・レジスタ(ACR)をそれぞれ表している(第2図参照)。

このように、チャネル形式でない入出力命令語体系のデータ処理装置では、1つの入出力動作(第1図ではシーク、リード動作)を行うために

複数個(第1図では7個)の入出力命令を発行する必要がある。したがって、従来、ターゲット・マシンがチャネル形式でない入出力命令語体系を有する場合のエミュレーションを、チャネル形式の入出力命令語体系を有するホステイブ・マシンで行う場合には、WIO命令が発行される度にこれをインタプリットするインタフエース・プログラムが起動されて、いくつかのWIO命令のセットを、チャネル形式の入出力命令または新オペレーティング・システムに対する入出力マクロ命令に置換えている。

この場合、WIO命令のインタプリットに要するソフトウェアのオーバーヘッドが非常に大きくなることは明らかである。エミュレーションされるプログラムがバッチ処理を行う場合のように、あまり高速性が要求されないときには、上記の方式によっても大きな問題は生じないが、エミュレーションされるプログラムがオンライン・システムのプログラムのように高速レスポンスを要求し、かつ入出力動作が頻繁に行われるような場合には、

性能の見地からこれを採用することはできない。
〔発明の目的〕

本発明の目的は、このような従来の問題点を改善し、入出力命令のエミュレーションを効率よく実行して、頻繁に入出力命令が発行されるオンライン・プログラムのエミュレーションを少ないオーバーヘッドで実現できるエミュレーション方式を提供することにある。

〔発明の概要〕

上記目的を達成するため、本発明のエミュレーション方式は、命令語体系の異なるデータ処理装置のプログラムをエミュレーションするマイクロプログラム制御のデータ処理装置において、被エミュレーション・プログラム中のスーパーバイザ・コール命令または直接オペレーティング・システムに到達する命令を検出し、該命令のオペランド値により入出力制御マクロ命令か否かを判定して、入出力制御マクロ命令であるときには、エミュレーションをコントロールするプログラムに割込みを発生し、上記以外の命令であるときには、被エ

ミュレーション・プログラムのオペレーティングシステムに制込みを発生することに特徴がある。
(発明の実施例)

第3図は、本発明の一実施例を示すエミュレーション時のプログラム構造を表す図である。

第3図において、10はホスト・マシンのオペレーティング・システム、11~13はホスト・マシンのユーザ・プログラム(UP)、20はエミュレーション・ジョブ(つまり、ターゲット・マシンのプログラムによるジョブ)、21はECP(Emulation Control Program)と呼ばれるインタフェース・プログラム、22はエミュレーションされるプログラム、23はターゲットマシンのオペレーティング・システム、24はターゲット・マシンのユーザ・プログラム(UP)である。

本発明においては、ホスト・マシンのオペレーティング・システム10とターゲット・マシンのプログラム22の間にECP21が設けられ、このECP21がエミュレーションされるプロ

ラム22に対しEXL命令(Execute Local)を発行してエミュレーションの起動を行い、またターゲット・マシンのユーザ・プログラム24内で発行される入出力制御マクロ命令を、ホスト・マシンのオペレーティング・システム10に対する入出力制御マクロ命令に変換する。

第4図は、EXL命令のフォーマット図である。

EXL命令は、プログラム処理状態をエミュレータが動作するモードに切換えて、PSW(Program Status Word)や汎用レジスタ等を設定するものである。そのフォーマットは、第4図に示すように、命令コード部(EXL)とB₂部(4ビットでベース・レジスタとなる汎用レジスタを指定する)とD₂部(12ビットで2進数値を表す)から構成される。この命令が出されると、B₂部で指定された汎用レジスタの内容をD₂部に加えて第2オペランド・アドレスとする。第2オペランド・アドレスは、主記憶装置のアドレスを指しており、ここに格納されている第2オペランドはローカル実行リストである。ローカル実行リス

には、エミュレータの動作に必要なPSW、汎用レジスタ、その他の情報が含まれており、その長さは40バイトである。EXL命令を実行すると、CPUはPSW、汎用レジスタの値をローカル実行リストの内容を用いて設定し、その後ローカル実行モード(エミュレータ動作モード)となる。

本発明のエミュレーション方式では、被エミュレーション・プログラム22内のユーザ・プログラム24から発行される入出力制御命令は、被エミュレーション・プログラム22のオペレーティング・システム23の入出力制御マクロ命令の形で発行されること、そして通常はスーパーバイザ・コール命令(SVC命令)のようなオペレーティング・システムへの制込み命令であること、に着目し、従来のように入出力命令を機械語のレベルでエミュレーションするのではなく、マクロ命令のレベルでエミュレーションを行うようにした。

これにより、ターゲット・マシンのユーザ・プログラム24内で発行された入出力制御マクロ命令を、ホスト・マシンのオペレーティング・

システム10に対する入出力制御マクロ命令に変換する操作が容易になるとともに、複数個の入出力命令に対してその度ごとに処理を行う必要がないので、少ないオーバーヘッドでエミュレーションを実行することができる。

すなわち、本発明では、エミュレーションを実行するマイクロプログラムにおいて、被エミュレーション・プログラム22中にスーパーバイザ・コール命令を検出したときには、その命令のオペランド、つまりSVCコードをあらかじめ用意されたテーブル(ローカル実行リストのエントリから取得される)により検索し、入出力制御マクロ命令によるスーパーバイザ・コール命令か否かの判定を行い、入出力制御マクロ命令によるスーパーバイザ・コール命令であつたときには、被エミュレーション・プログラム22のオペレーティング・システム23に制込みを発生せず、一旦エミュレーション・モードを解除し、エミュレーション・コントロール・プログラムと呼ばれ、ホスト・モードで動作するインタフェース・プログラム

(ECP21)に割込みを発生させる。エミュレーション・コントロール・プログラム(ECP21)では、被エミュレーション・プログラム22から発行された入出力制御マクロ命令を解析し、ネーティブ・モード・オペレーティング・システム10の入出力制御マクロ命令に変換して発行する。

第5図は、本発明の実施例を示すエミュレーション方式の処理フローチャートおよびローカル実行リストの説明図である。

第5図(a)には、入出力制御マクロ変換を行うまでのエミュレーション・マイクロプログラムの動作フローが示されている。

エミュレーション・マイクロプログラムがターゲット・マシンの機械語を逐次実行していく途中で、スーパーバイザ・コール命令(SVC命令)の命令コードを検出すると、第4図(b)に示すようなエミュレーション起動時に設定されているローカル実行リスト(エミュレーション・モード制御テーブル)41を参照して、その中の1つのエント

リであるSVCジャンプ・テーブル先頭アドレスを取得する(ステップ31, 33)。取得した先頭アドレスとジャンプ・テーブル42のSVCコード(SVC命令のオペランド値であつて、0~255の値をとる)に対応する番地1バイトにセットされている値を参照する。このSVCジャンプ・テーブル42の値は、ターゲット・マシン・オペレーティング・システム23のSVCコードを判定して、あらかじめコーディングされているものであり、入出力制御マクロ命令に関するSVCコードに対応するエントリには $(FF)_{16}$ がセットされ、その他のエントリには $(FF)_{16}$ 以外の値がセットされている。エミュレーション・マイクロプログラムは、テーブル42のエントリの値を判別し、そのSVCコードに対応するエントリの値が $(FF)_{16}$ でない場合には、ターゲットマシンのオペレーティング・システム23内で処理可能なSVC命令であると判定して、ターゲット・マシンのアーキテクチャで定められたSVC命令の動作を行うようにエミュレーションする。

(ステップ34, 35)。エントリの値が $(FF)_{16}$ の場合には、入出力制御マクロ命令に関するSVC命令と判定し、現在のエミュレーション・モードの状態を示すPSWを、第5図(b)に示すローカル実行リスト41内のPSW退避エリアに退避すると同時に、ECP21のターゲット・マシンの入出力制御マクロ命令の変換を行うための入口アドレスを示すECP割込み新PSWをPSWにロードし、エミュレーション・モードを解除して、ECP21に制御を渡す(ステップ34~39)。

このようにして、エミュレーション・マイクロプログラムの制御により、入出力制御マクロ命令に関するSVC命令が検出され、ECP21に制御が渡される。

ECP21は、ローカル実行リスト41中のローカルPSWを参照し、SVC命令のアドレスを知ることができるので、SVC命令に付随する入出力制御に関連するパラメータ群を認識することができ、容易にネーティブ・マシンのオペレーティング・システム10に対する入出力マクロ命令に

変換することが可能となる。

また、これにより、ターゲット・マシンのオペレーティング・システム23の入出力制御マクロ命令処理ルーチンは、全く動作しなくてもすむので、ソフトウェアのオーバーヘッドは非常に少なくなる。

(発明の効果)

以上説明したように、本発明によれば、異なるアーキテクチャを有するデータ処理装置のエミュレーションにおいて、スーパーバイザ・コール命令によりターゲット・マシンのユーザ・プログラムから発行される入出力制御マクロ命令を判別し、これをインタフエース・プログラムに連絡する機能を備えたので、機械語レベルでエミュレーションすることなく、マクロ命令レベルでエミュレーションすることができ、きわめて簡単にネーティブ・マシンの入出力制御マクロ命令に変換することが可能となる。したがって、少ないオーバーヘッドによりエミュレーションできるので、頻繁に入出力命令を発行するオンライン・プログラムのエ

ミュレーションに好適である。

第 1 図

4. 図面の簡単な説明

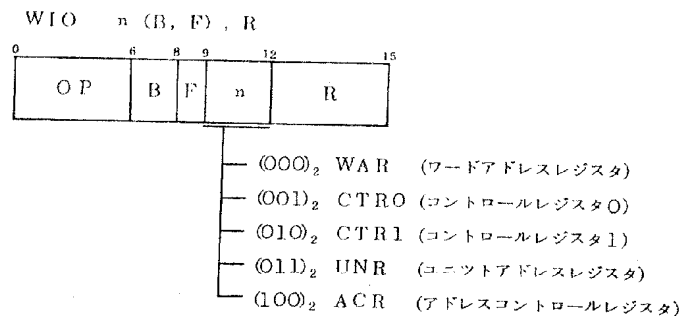
第1図および第2図はチャネル形式でない入出力命令体系を有する処理装置の入出力命令群の説明図と1命令のフォーマット図、第3図は本発明の一実施例を示すエミュレーション時のプログラム構造を表した図、第4図はEXL命令のフォーマット図、第5図は、本発明の一実施例を示すエミュレーション方式の処理フローチャートとローカル実行リストの説明図である。

10：ホータイプ・マシンのオペレーティングシステム、11～13：ホータイプ・マシンのユーザ・プログラム、20：エミュレーション・ジョブ、21：インタフェース・プログラム（ECP）、22：被エミュレーション・プログラム、23、ターゲット・マシンのオペレーティングシステム、24：ターゲット・マシンのユーザ・プログラム。

特許出願人 株式会社 日立製作所
代理人 弁理士 磯村 雅 俊

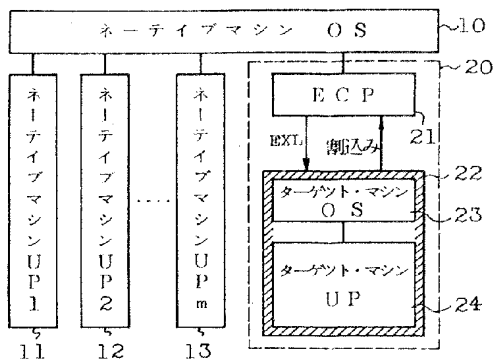
- (a)
- ① WIO UNR (B, 0), R1 ユニットアドレスの指定
 - ② WIO CTRL (B, 0), R2 シークアドレスの指定
 - ③ WIO CTRO (B, 1), R3 シーク動作の指定, 起動
- (b)
- ① WIO ACR (B, 0), R1 リードアドレスの指定
 - ② WIO WAR (B, 0), R2 リードワード数の指定
 - ③ WIO CTRL (B, 0), R3 リードワード数の指定
 - ④ WIO CTRO (B, 1), R4 リード動作の指定, 起動

第 2 図



第 5 図

第 3 図



第 4 図

